Grafy eulerowskie i hamiltonowskie

dr inż. Bartłomiej Pawlik

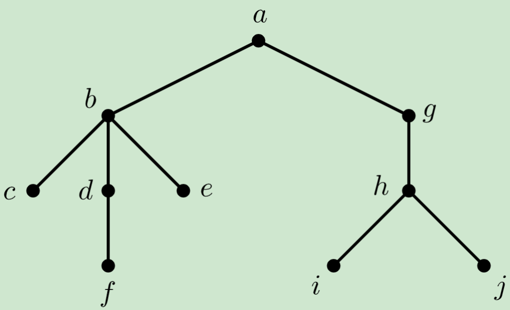
28 lipca 2024

Drzewa przeszukiwań

- **Metoda przeszukiwania wszerz** (**breadth-first search**, **BFS**)
  Odwiedzamy wszystkich sąsiadów aktualnego wierzchołka, zanim przejdziemy do następnego wierzchołka.
- **Metoda przeszukiwania w głąb** (**deapth-first search**, **DFS**)
  Po odwiedzeniu sąsiada v_{k+1} wierzchołka v_k, przechodzimy do nieodwiedzonego sąsiada v_{k+2} wierzchołka v_{k+1} albo — w przypadku braku nieodwiedzony sąsiadów — cofamy się do wierzchołka v_k i powtarzamy.

W *BFS* zanim zagłębimy się bardziej w drzewo, zaglądamy do tak wielu wierzchołków jak to możliwe, a w *DFS* zagłębiamy się tak daleko, jak tylko jest to możliwe, zanim zajrzymy do innych wierzchołków sąsiednich.

Przykład



Przykładowe wyniki algorytmów przeszukiwania wszerz i w głąb, zaczynając od wierzchołka a:

$$BFS: \ a, b, g, c, d, e, h, f, i, j$$

$$DFS: \ a, b, c, d, f, e, g, h, i, j$$

Jakie wyniki może zwrócić algorytm, gdy zaczniemy od wierzchołka d?

Definicja

Niech $G = (V(G), E(G))$ będzie grafem i niech $w : E(G) \to \mathbb{R}$ będzie funkcją. Parę (G, w) nazywamy *grafem ważonym*.

Innymi słowy, grafem ważonym nazywamy graf, w którym każdej krawędzi przypisana jest liczba rzeczywista (może ona reprezentować odległość między wierzchołkami, przepustowość sieci, ilość interakcji itd.).

Szukanie najkrótszej drogi - algorytm Dijkstry

Algorytm służy do wyszukiwania najkrótszej drogi od danego wierzchołka do pozostałych w grafie ważonym bez pętli, w którym wagi są liczbami <u>nieujemnymi</u>.

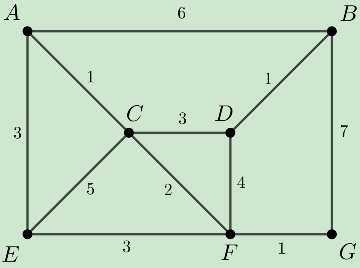
Niech $V(G) = \{0, 1, 2, \ldots, n-1\}$. Będziemy szukać najkrótszej drogi od wierzchołka 0 do pozostałych.
Krok 1: $Q := \{0, 1, \ldots, n\}$, $S := \emptyset$,
$d(0) := 0$, $p(0) := 0$ oraz $d(v) := \infty$, $p(v) := \infty$ dla $1 \leqslant v \leqslant n$.
Krok 2: Dopóki $Q \neq \emptyset$ powtarzaj:

1. Wybierz $v \in Q$ o najmniejszej wartości $p(v)$.
2. $Q := Q/\{v\}$, $S := S \cup \{v\}$.
3. Dla każdego wierzchołka u sąsiedniego z v dokonaj **relaksacji**, tzn. jeśli $d(u) > d(v) + w(v, u)$, to $d(u) := d(u) + w(v, u)$ oraz $p(u) = v$.

Krok 3: Minimalna długość drogi od 0 do v to $d(v)$. Jeśli $d(v) = \infty$ to nie istnieje droga od 0 do $d(v)$ (to jest możliwe w digrafie).

Przykład

W poniższym grafie znaleźć najkrótsze drogi z wierzchołka B do pozostałych wierzchołków korzystając z algorytmu Dijkstry.



	A	C	D	E	F	G
B_0	6_B	∞	1_B	∞	∞	7_B
BD_1	6_B	4_D	$-$	∞	5_D	7_B
BDC_4	5_C	$-$	$-$	9_C	5_D	7_B
BDF_5	5_C	$-$	$-$	8_F	$-$	6_F
$BDCA_5$	$-$	$-$	$-$	8_F	$-$	6_F
$BDFG_6$	$-$	$-$	$-$	8_F	$-$	$-$
$BDFE_8$	$-$	$-$	$-$	$-$	$-$	$-$

W pierwszej kolumnie zapisane są najkrótsze drogi (w indeksie każdej z nich jest waga).
Elementy tabeli to wagi dróg z wierzchołka B do danego wierzchołka (w indeksach są wierzchołki bezpośrednio poprzedzające dany).
Symbol „$-$" oznacza, że do danego wierzchołka już określono najkrótszą drogę.
W sytuacji, gdy w wierszu mamy więcej niż jedną drogę o najmniejszej wadze (5_C do A i 5_D do F w wierszu czwartym), wybieramy dowolną z nich.

Zauważmy, że problem najkrótszej drogi można rozwiązać również algorytmem *brute force* — wystarczy określić wagę wszystkich ścieżek[1] między rozważanymi wierzchołkami i wybrać najmniejszą. Jednak już nawet dla grafów o niewielkiej liczbie wierzchołków ten algorytm jest dużo mniej optymalny.

Zauważmy jednak, że znalezienie jakiejkolwiek ścieżki między rozważanymi wierzchołkami daje w oczywisty sposób <u>ograniczenie górne</u> na rozwiązanie problemu najkrótszej drogi.

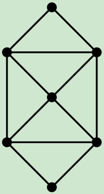
[1] Oczywiście najkrótsza droga musi być ścieżką.

Grafy eulerowskie

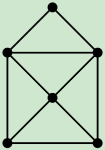
Definicja

- Jeżeli w grafie G istnieje cykl niewłaściwy d przechodzący przez każdą krawędź grafu G dokładnie jeden raz, to d nazywamy **cyklem Eulera**, a graf G — **grafem eulerowskim**.
- Jeżeli graf G nie jest grafem eulerowskim i istnieje ścieżka d przechodząca przez każdą krawędź grafu G dokładnie jeden raz, to G nazywamy **grafem jednobieżnym** (**półeulerowskim**).

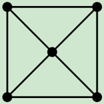
Przykład



G_1



G_2



G_3

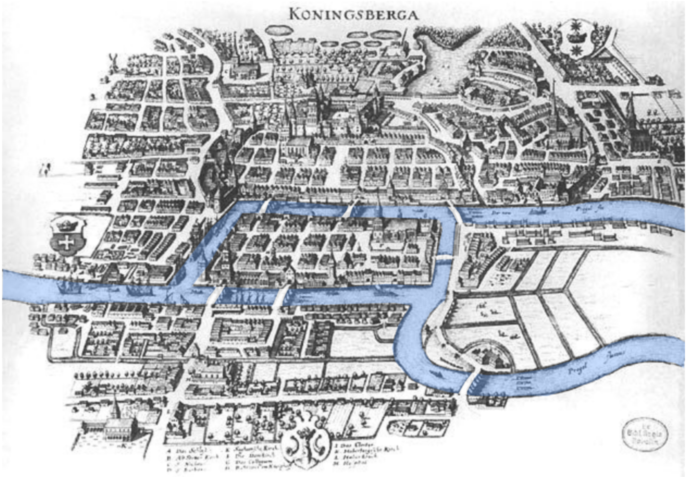
G_1 — graf eulerowski

G_2 — graf jednobieżny

G_3 — graf, który nie jest ani eulerowski, ani jednobieżny

Problem mostów królewieckich

Czy można przejść dokładnie jeden raz przez każdy z siedmiu mostów przedstawionych na poniższym rysunku i powrócić do punktu wyjścia?



[Źródło: *https://pl.wikipedia.org/wiki/Królewiec#/media/Plik:Image-Koenigsberg,_Map_by_Merian-Erben_1652.jpg*, 13.05.2024]

Lemat 1

Jeżeli $\delta(G) \geqslant 2$, to graf G zawiera cykl.

Dowód.

Jeżeli G zawiera pętle lub krawędzie wielokrotne, to lemat jest trywialny. Załóżmy zatem, że G jest grafem prostym i że $v \in G$ jest dowolnym wierzchołkiem G. Tworzymy trasę

$$v \longrightarrow v_1 \longrightarrow v_2 \longrightarrow \dots$$

tak aby v_{k+1} był sąsiadem v_k różnym od v_{k-1} (zawsze jest to możliwe ze względu na założenie, że stopień każdego wierzchołka wynosi co najmniej 2). Graf G ma skończenie wiele wierzchołków, więc po skończonej liczbie kroków na trasie musi pojawić się wierzchołek v_K, który pojawił się na niej już wcześniej. Fragment trasy

$$v_K \longrightarrow \dots \longrightarrow v_K$$

jest szukanym cyklem. $\square$

Twierdzenie (Euler, 1741)

Niech G będzie grafem spójnym. Graf G jest eulerowski wtedy i tylko wtedy, gdy każdy wierzchołek ma stopień parzysty.

Dowód. *(1/2)*

($\Rightarrow$)
Niech d będzie cyklem Eulera w grafie G i niech $v \in V(G)$. Każde przejście cyklu d przez wierzchołek v zwiększa udział krawędzi należących do tego cyklu w stopniu wierzchołka v o 2. Każda krawędź występuje w grafie G dokładnie raz, więc stopień każdego wierzchołka musi być liczbą parzystą.

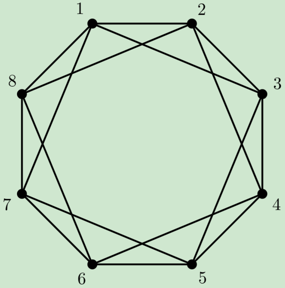
Dowód *(2/2)*

($\Leftarrow$)

Dowód indukcyjny względem liczby krawędzi grafu G. G jest grafem spójnym, w którym stopień każdego wierzchołka jest liczbą parzystą, więc $\delta(G) \geqslant 2$. Zatem, na mocy Lematu 1, w G istnieje cykl c.

- Jeżeli c zawiera każdą krawędź grafu G, to twierdzenie jest udowodnione.
- Jeżeli c nie zawiera każdej krawędzi grafu G, to usuwamy z grafu G wszystkie krawędzie należące do cyklu c, otrzymując graf $G \backslash \{c\}$, który ma mniej krawędzi niż G. Z założenia indukcyjnego każda składowa spójności grafu $G \backslash \{c\}$ posiada cykl Eulera. Ze spólności G wynika, że każda składowa spójności grafu $G \backslash \{c\}$ ma co najmniej jeden wierzchołek wspólny z cyklem c. Zatem w grafie G można stworzyć cykl Eulera poprzez połączenie cyklu c z cyklami Eulera wszystkich składowych spójności grafu $G \backslash \{c\}$ poprzez wierzchołki wspólne.

Przykład



Na powstawie powyższego twierdzenia Eulera określić czy dany graf jest eulerowski lub jednobieżny.

- Jeżeli jest eulerowski, to wskazać przykładowy cykl Eulera.
- Jeżeli jest jednobieżny to wskazać przykładową ścieżkę Eulera.

Dany graf jest eulerowski, ponieważ stopień każdego wierzchołka jest liczbą parzystą. Przykładowy cykl Eulera to

$$(1 \to 2 \to 8 \to 1) \to (3 \to 4 \to 2 \to 3) \to (5 \to 6 \to 4 \to 5) \to$$

$$\to (7 \to 8 \to 6 \to 7) \to 1.$$

Wniosek

Niech G będzie grafem spójnym.

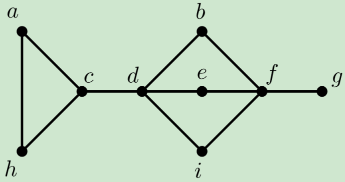
- Graf G jest grafem eulerowskim wtedy i tylko wtedy, gdy jego zbiór krawędzi można podzielić na rozłączne cykle.
- Graf G jest grafem jednobieżnym wtedy i tylko wtedy, gdy ma dokładnie dwa wierzchołki nieparzystych stopni.

Znajdowanie cyklu Eulera

Definicja

Mostem nazywamy tę krawędź w grafie skończonym G, której usunięcie powoduje zwiększenie liczby spójnych składowych grafu G.

Przykład



Jedynymi mostami w powyższym grafie są krawędzie $\{c, d\}$ oraz $\{f, g\}$.

Znajdowanie cyklu Eulera - algorytm Fleury'ego

Zaczynając od dowolnego wierzchołka, tworzymy cykl Eulera dodając do niego kolejne krawędzie w taki sposób, że dodajemy do cyklu most tylko wtedy, gdy nie ma innej możliwości.

Przykład

Wyznaczyć cykl Eulera w grafie $K_{2,4}$.

Problem chińskiego listonosza

Sformułowanie

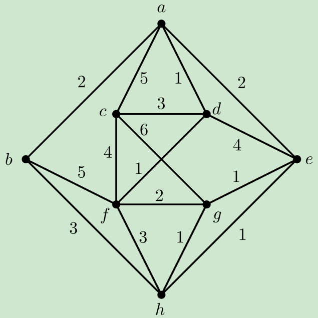
Znalezienie cyklu (niewłaściwego) zawierającego każdą krawędź danego grafu co najmniej raz i mającego jak najmniejszy koszt (czyli liczbę krawędzi lub, w przypadku grafu ważonego, sumę wag krawędzi).

Rozwiązanie problemu chińskiego listonosza

- Jeżeli graf G jest eulerowski to znajdujemy dowolny cykl Eulera.
- Jeżeli graf nie jest eulerowski to dublujemy niektóre krawędzie, aby otrzymać graf eulerowski i wtedy szukamy ścieżki Eulera. Aby otrzymać najoptymalniejsze (najkrótsze) ścieżki
  - dla grafów nieważonych wystarczy przeszukać graf wszerz,
  - dla grafów z nieujemnymi wagami można skorzystać np. z algorytmu Dijkstry,
  - dla grafów z dowolnymi wagami można skorzystać np. z algorytmu Bellmana-Forda.

Przykład

Rozwiąż problem chińskiego listonosza dla poniższego grafu. Jaki jest koszt cyklu stanowiącego rozwiązanie?



Wierzchołki b i f są jedynymi wierzchołkami nieparzystego stopnia, więc dany graf jest jednobieżny. Najkrótsza droga z b do f to

$$b \to a \to d \to f,$$

więc w grafie dublujemy krawędzie

$$\{a, b\},\ \{a, d\} \text{ oraz } \{d, f\},$$

dzięki czemu uzyskaliśmy graf eulerowski.

Przykładowym cyklem Eulera w nowym grafie (i zarazem rozwiązaniem problemu chińskiego listonosza) jest

$$a \to e \to h \to b \to a \to d \to f \to b \to a \to d \to$$

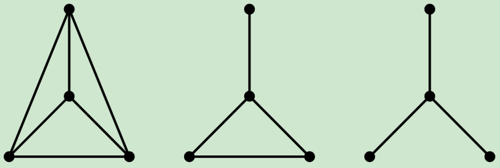
$$\to e \to g \to h \to f \to d \to c \to g \to f \to c \to a,$$

którego koszt wynosi 48.

Definicja

- Jeżeli w grafie G istnieje cykl h przechodzący przez każdy wierzchołek grafu G dokładnie jeden raz, to h nazywamy **cyklem Hamiltona**, a graf G — **grafem hamiltonowskim**.
- Jeżeli graf G nie jest grafem hamiltonowskim i istnieje ścieżka h przechodząca przez każdy wierzchołek tego grafu dokładnie jeden raz, to G nazywamy **grafem trasowalnym** (**półhamiltonowskim**).

Przykład



G_1 G_2 G_3

G_1 — graf hamiltonowski

G_2 — graf trasowalny

G_3 — graf, który nie jest ani hamiltonowski, ani trasowalny

Uwaga!

Powszechnie przyjęte nazewnictwo: **graf półeulerowski**, **graf półhamiltonowski** jest trochę niefortunne, bo sugeruje, że każdemu z tych grafów dużo brakuje do bycia „pełnymi" grafami eulerowskimi lub hamiltonowskimi, podczas gdy w każdym przypadku wystarczy w tym celu do grafu dodać tylko jedną krawędź.

Twierdzenie (Ore, 1960)

Jeżeli graf prosty G ma n wierzchołków ($n \geqslant 3$) oraz

$$\deg(u) + \deg(v) \geqslant n$$

dla każdej pary wierzchołków niesąsiednich u i v, to graf G jest hamiltonowski.

Dowód. (*1/2*)

Załóżmy nie wprost, że dla ustalonego $n \geqslant 3$ istnieją grafy niehamiltonowskie spełniające założenia rozpatrywanego twierdzenia. Niech G będzie takim grafem z jak największą liczbą krawędzi — jeżeli do G dołączymy jedną krawędź to otrzymamy graf hamiltonowski.

Dowód. (2/2)

W G istnieje co najmniej jedna para wierzchołków, które nie są połączone krawędzią — bez straty ogólności przyjmijmy że są to wierzchołki v_1 i v_n. Z faktu, że po dodaniu krawędzi otrzymalibyśmy cykl Hamiltona wynika, że w grafie G istnieje droga

$$v_1 \to v_2 \to \ldots \to v_{n-1} \to v_n.$$

Z założenia wiemy, że

$$\deg v_1 + \deg v_n \geqslant n,$$

co oznacza że istnieje indeks k taki, że v_{k-1} jest sąsiadem v_n i że v_k jest sąsiadem v_1. Jednak teraz cykl

$$v_1 \to \ldots \to v_{k-1} \to v_n \to \ldots \to v_k \to v_1$$

jest cyklem Hamiltona, co daje nam szukaną sprzeczność.

Twierdzenie (Dirac, 1952)

Jeżeli minimalny stopień grafu G jest nie mniejszy niż połowa liczby wierzchołków tego grafu:

$$\delta(G) \geqslant \frac{|V(G)|}{2},$$

to G jest grafem hamiltonowskim.

Dowód.

Niech $u, v \in V(G)$ i niech $|V(G)| = n$. Każdy wierzchołek grafu G ma stopień nie mniejszy niż $\frac{n}{2}$, więc

$$\deg(u) + \deg(v) \geqslant \frac{n}{2} + \frac{n}{2} = n.$$

Zatem w szczególności sumy stopni wszystkich par niesąsiednich wierzchołków są nie mniejsze niż n, więc z twierdzenia Orego otrzymujemy, że G jest grafem hamiltonowskim. $\square$

Problem komiwojażera

Podamy dwie wersje problemu:

- Mając daną listę miast i odległości między tymi miastami, znaleźć najkrótszą drogę przechodzącą przez wszystkie miasta (przez każde tylko raz) i powracającą do punktu wyjścia.
- Znaleźć najoptymalniejszy cykl Hamiltona w ważonym grafie pełnym.

Przykładowe rozwiązania

- *Brute force*: Znajdujemy wszystkie cykle Hamiltona i wybieramy najodpowiedniejszy.
- *Algorytm najbliższego sąsiada*: Zaczynamy od dowolnego wierzchołka i poruszamy się zawsze wzdłuż krawędzi o najmniejszych wagach. (jest to rozwizanie przybliżone, średnio o 25% gorsze od optymalnego).